Parallel BioScape: A Stochastic and Parallel Language for Mobile and Spatial Interactions [*]

Adriana Compagnoni
Department of Computer Science
Stevens Institute of Technology

Mariangiola Dezani–Ciancaglini
Dipartimento di Informatica
Università di Torino

Paola Giannini
Dipartimento di Informatica
Università del Piemonte Orientale

Karin Sauer
Department of Biological Sciences
Binghamton University

Vishakha Sharma
Department of Computer Science
Stevens Institute of Technology

Angelo Troina
Dipartimento di Informatica
Università di Torino

BioScape is a concurrent language motivated by the biological landscapes found at the interface of biology and biomaterials [5]. It has been motivated by the need to model antibacterial surfaces, biofilm formation, and the effect of DNAse in treating and preventing biofilm infections. As its predecessor, SPiM [12], BioScape has a sequential semantics based on Gillespie's algorithm [7], and its implementation does not scale beyond 1000 agents. However, in order to model larger and more realistic systems, a semantics that may take advantage of the new multi-core and GPU architectures is needed. This motivates the introduction of parallel semantics, which is the contribution of this paper: Parallel BioScape, an extension with fully parallel semantics.

Process algebras have been successfully used in the modeling of biological systems, see [14, 4, 1], where they are particularly attractive, because of their ability to accommodate new objects and new behavioral attributes as the complex biological system becomes better understood. However, most of the modeling languages lack adequate support for the design of systems in which to study complex interactions involving both spatial properties, movements in three-dimensional space, and stochastic interactions. Recently, new spatial modeling languages allowing explicit description of temporal spatial dynamics of biochemical processes have been proposed (SpacePi [8], DCA [17], $L\Pi$ [16], Stochsim [10]). Other agent-based platforms [9] include C-Immsim [15, 11] and PathSim visualizer [13]. However, few of them support individual based, continuous motion, and continuous space stochastic simulation [3], which are important features for modeling temporal spatial dynamics of biochemical processes accurately. To address this problem in previous work we introduced BioScape [5], a language incorporating both stochasticity and 3D spatial attributes.

Gillespie's algorithm produces two outputs in each iteration: 1) the next reaction R to be executed and 2) a slice of time t to advance the simulation clock. Since many reactions, including many instances of the same reaction, may be available, the slice of time t does not correspond to the time that R would take, but an amount of time proportional to the time it would take to execute all available reactions. In contrast, the parallel semantics will execute all available reactions, not just one instance of one reaction R, and the first challenge is then how to calculate simulated time. Reaction times can vary substantially, for example, some prokaryotic cell mitosis takes ten minutes, some plant cell mitosis takes about half an hour, while some animal cell mitosis takes about three hours. If we trigger all reactions together, how do

[*] Extended Abstract. This research is funded by the BioBITs Project (*Converging Technologies* 2007, area: Biotechnology–ICT), Regione Piemonte.

Gabriel Ciobanu (Ed.): Membrane Computing and
Biologically Inspired Process Calculi (MeCBIC 2012)
EPTCS 100, 2012, pp. 101–106, doi:10.4204/EPTCS.100.7

$$
\begin{array}{rllll}
P,Q ::= & 0 \quad\mid\; X(\overline{u}) \quad\mid\; P \mid Q \quad\mid\; (\nu a@\mathtt{r},\mathtt{rad}).P & \quad D ::= & \emptyset \quad\mid\; D, X(\overline{x}) = M^{\xi,\omega,\sigma} & \mathtt{FV}(M) \subseteq \overline{x} \\
M ::= & \pi.P\,[+\,M] & \quad u,v ::= & a \mid b \mid \cdots \mid x \mid y \mid \cdots & \\
\pi ::= & \mathtt{delay@r} \quad\mid\; !u(v) \quad\mid\; ?u(x) \quad\mid\; \mathtt{mov} & \quad E ::= & \emptyset \quad\mid\; E, a@\mathtt{r},\mathtt{rad} &
\end{array}
$$

Figure 1: Syntax

we advance the simulation clock? The solution we propose here consists of annotating each product of a reaction with a timer indicating how long that reaction will take.

For example, if $Cell \rightarrow_{30} Cell \mid Cell$ means that a $Cell$ takes 30 minutes to split, through mitosis, into two daughter cells, then we will annotate the two daughter cells as $\{\!\{Cell\}\!\}^{30}$ and $\{\!\{Cell\}\!\}^{30}$. As time lapses, the timer will be reduced, and when reaching $\{\!\{Cell\}\!\}^{0}$, both cells will be available for new reactions.

In Fig. 1 we define the syntax of the calculus, which slightly simplifies the syntax of [5] in order to avoid decorating semantic processes with shapes, as defined at page 103.

We assume a set of channel names, denoted by a, b, and a set of variables, denoted by x, y, with subscripts or superscripts, if needed. As usual, $\overline{a}$ is $a_1,\ldots,a_n$, and similar for $\overline{x}$. The empty process is 0. By $X(\overline{u})$ we denote an instance of the entity defined by X. The actual parameters of the instance may be either channel names or variables, in case the instance occurs in a definition. The process $P \mid Q$ is the parallel composition of processes P and Q. By $(\nu a@\mathtt{r},\mathtt{rad}).P$ we define the channel name a with two parameters $\mathtt{r}$ and $\mathtt{rad} \in \mathbb{R}_{\geq 0}$ within process P; the parameter $\mathtt{r}$ is the stochastic rate for communications through channel a and $\mathtt{rad}$ is the communication radius. The radius is the maximum distance between processes in order to communicate through channel a, and the reaction rate determines how long it takes for two processes to react given that they are close enough to communicate.

The *heterogeneous* choice is denoted by M, where $\pi.P\,[+\,M]$ means $\pi.P \;\mid\; \pi.P + M$. Choices may have reaction branches and movement branches. The reaction branches are probabilistic (stochastic), since reactions are subject to kinetic reaction rates, while the movement branches are non-deterministic, since the movement of instances of entities is always enabled, provided there is enough space. The prefix π denotes the action that the process $\pi.P$ can perform. The prefix $\mathtt{delay@r}$ is a spontaneous and unilateral reaction of a single process, where $\mathtt{r}$ is the stochastic rate. The prefix $!u$ denotes output, and the prefix $?u$ denotes input. The prefix $\mathtt{mov}$ moves processes in space according to their diffusion rate (ω) (see below). We use standard syntactic abbreviations such as π for $\pi.0$.

We denote by D a global list of definitions. The equality $X(\overline{x}) = M^{\xi,\omega,\sigma}$ defines entity X with formal parameters $\overline{x}$, to be the choice M with geometry ξ,ω,σ, specifying a movement space ξ, a step ω, and a shape σ. The choice M describes the behavior of X with a choice of prefixed processes. The selection of one of the choices depends not only on the available interactions with other processes, but also on the available space. The movement space ξ is a set of point coordinates in the global coordinate system defining a volume. Intuitively, X can move within ξ. The step $\omega \in \mathbb{R}_{\geq 0}$, is the distance that X can stir in a movement, and it corresponds to the diffusion rate of X; σ is the three-dimensional shape (sphere, cube, etc.) of X. The movement space for the empty process 0 is everywhere, the global space, and its movement step is 0 by default. The entity variable X can be defined at most once in D, and the free variables of P, denoted by $\mathtt{FV}(P)$, must be a subset of the variables $\overline{x}$. We also write $X(\overline{x}) = (\pi.\pi'.P)^{\xi,\omega,\sigma}$ as short for $X(\overline{x}) = (\pi.Y(\overline{x}))^{\xi,\omega,\sigma}$ and $Y(\overline{x}) = (\pi'.P)^{\xi,\omega,\sigma}$.

We use E to range over environments of channel name declarations. By $a@\mathtt{r},\mathtt{rad}$ we declare channel name a with reaction rate $\mathtt{r}$ and reaction radius $\mathtt{rad}$. A channel name a appears at most once in E.

Consider the following simple example of a bacterium `Bac`, that can either move or divide into two daughter cells. `Bac` is defined with movement space `movB`, movement step `stepB`, and shape `shapeB`. Intuitively, bacteria can move within `movB`, with non-deterministic steps of length `stepB`, and the shape

S.LOC
$$\frac{P \equiv Q}{\{P\}_\mu \equiv \{Q\}_\mu}$$

S.LOC.NU
$$\frac{}{(\nu a@\mathtt{r},\mathtt{rad}).\{P\}_\mu \equiv \{(\nu a@\mathtt{r},\mathtt{rad}).P\}_\mu}$$

S.LOC.PAR
$$\frac{\mu_1(\mathtt{shape}(P)) \cup \mu_2(\mathtt{shape}(Q)) = \mu(\mathtt{shape}(P \mid Q))}{\{P\}_{\mu_1} \mid \{Q\}_{\mu_2} \equiv \{P \mid Q\}_\mu}$$

S.NU.COM
$$\frac{}{(\nu a@\mathtt{r},\mathtt{rad}).(\nu b@\mathtt{r}',\mathtt{rad}').A \equiv (\nu b@\mathtt{r}',\mathtt{rad}').(\nu a@\mathtt{r},\mathtt{rad}).A}$$

S.NU.PAR
$$\frac{a \notin \mathtt{fn}(B)}{((\nu a@\mathtt{r},\mathtt{rad}).A) \mid B \equiv (\nu a@\mathtt{r},\mathtt{rad}).(A \mid B)}$$

Figure 2: Structural Equivalence of Spatial Configurations

`shapeB` is at all times contained within `movB`. The prefix `mov` represents a non-deterministic movement of length `stepB`, whereas `delay@1.0.(Bac() | Bac())` represents mitosis, the division of a bacterium into two daughter cells: `Bac() | Bac()`, and the `delay@1.0` prefix is used to model the fact that division is not an instantaneous reaction.

$$\mathtt{Bac}() = (\mathtt{mov}.\mathtt{Bac}() + \mathtt{delay}@1.0.(\mathtt{Bac}()|\mathtt{Bac}()))^{\mathtt{movB},\mathtt{stepB},\mathtt{shapeB}}$$

A run-time system is represented by a parallel composition of entity instances (without free variables) each with its shape, and located in some positions of a global frame. We define the shape of processes inductively as follows:

$$\mathtt{shape}(0) = \emptyset \qquad \mathtt{shape}(X(\overline{a})) = \sigma \ \text{ if } X(\overline{x}) = M^{\xi,\omega,\sigma} \in D$$
$$\mathtt{shape}((\nu a@\mathtt{r},\mathtt{rad}).P) = \mathtt{shape}(P) \qquad \mathtt{shape}(P \mid Q) = \mathtt{shape}(P) \Cup \mathtt{shape}(Q)$$

where $\Cup$ gives a shape obtained by composing two shapes trough juxtaposition. For different applications we can choose suitable functions to realise $\Cup$, we only require $\Cup$ to be a commutative and associative operator, i.e. $\sigma_1 \Cup \sigma_2 = \sigma_2 \Cup \sigma_1$ and $(\sigma_1 \Cup \sigma_2) \Cup \sigma_3 = \sigma_1 \Cup (\sigma_2 \Cup \sigma_3)$.

We use μ to denote a map which applied to a shape locates it in the global space, by putting its barycentre at a fixed point, orienting the shape, and possibly modifying it. So $\mu(\mathtt{shape}(P))$ computes the space occupied by a process P in the global coordinate system. Processes may also share channels for communication. *Spatial configurations*, denoted by A, B, ... are defined as follows:

$$A, B \ ::= \{P\}_\mu \quad | \quad A \mid B \quad | \quad (\nu a@\mathtt{r},\mathtt{rad})..A$$

Structural equivalence on configurations is defined in Fig. 2, omitting the rules for associativity and commutativity of $|$ and $+$. Parallel composition has neutral element $\{0\}_\mu$ for any μ. Rule S.LOC uses the standard structural equivalence of Pi-calculus processes. The premise of rule S.LOC.PAR assures that the two equivalent processes occupy exactly the same space. In rule S.NU.PAR, `fn` is a function that returns the set of free channel names of a configuration.

The (parallel) operational semantics of BioScape is based on two *auxiliary* reduction relations: a stochastic relation, $E \vdash A \xrightarrow{\mathtt{r}} B$, for reactions such as synchronisation and delay, defined in Fig. 3, and a non-deterministic (non-stochastic) relation, $A \rightarrow B$, for geometric transformations, in our case movement, defined in Fig. 4. Notice that reduction axioms (SR.DELAY, SR.COM, NR.MOVE) only involve entities ($X(\overline{a})$), and entities evolve according to one of the choices in their definitions. In rules SR.DELAY, SR.COM and NR.MOVE, there is no check of whether the entities of the resulting process have enough space, since this check is done in the parallel reductionrules PR.STOC, and PR.MOVE of Fig. 5. In

SR.DELAY
$$\frac{X(\overline{x}) = (delay@\mathtt{r}.P\ [+M])^{\xi,\omega,\sigma} \in D}{E \vdash \{X(\overline{a})\}_\mu \xrightarrow{\mathtt{r}} \{P[\overline{a}/\overline{x}]\}_\mu}$$

SR.STR
$$\frac{A \equiv A' \qquad E \vdash A' \xrightarrow{\mathtt{r}} B' \qquad B' \equiv B}{E \vdash A \xrightarrow{\mathtt{r}} B}$$

SR.COM
$$\frac{X(\overline{x}) = (!a(b).P\ [+M])^{\xi,\omega,\sigma} \in D \quad Y(\overline{y}) = (?a(z).Q\ [+N])^{\xi',\omega',\sigma'} \in D \quad \mathtt{dis}(\mu,\mu') \leq \mathtt{rad}}{E, a@\mathtt{r},\mathtt{rad} \vdash \{X(\overline{c})\}_\mu \mid \{Y(\overline{d})\}_{\mu'} \xrightarrow{\mathtt{r}} \{P[\overline{c}/\overline{x}]\}_\mu \mid \{Q[\overline{d}/\overline{y}][b/z]\}_{\mu'}}$$

Figure 3: Stochastic Reduction Relation

$$\textsc{NR.Move}\ \frac{\mu' = \mathtt{translate}(\omega,\mu) \quad \mu'(\sigma) \subseteq \xi \quad X(\overline{x}) = (\mathtt{mov}.P\ [+\,M])^{\xi,\omega,\sigma} \in D}{\{X(\overline{a})\}_{\mu} \rightarrow \{P[\overline{a}/\overline{x}]\}_{\mu'}} \qquad \textsc{NR.Str}\ \frac{A \equiv A' \quad A' \rightarrow B' \quad B' \equiv B}{A \rightarrow B}$$

Figure 4: Non-stochastic Reduction Relation

particular, a stochastic (non-stochastic) redex is stuck, if there is not enough space for its reduct in the configuration. Therefore, the *evolution of systems in parallel BioScape produces configurations in which space is consistent*.

Fig. 3 defines the stochastic reduction relation of BioScape, $E \vdash A \xrightarrow{\mathtt{r}} B$, where $\mathtt{r}$ is the rate of the channel used for synchronization or delay. We write $\mathtt{dis}(\mu,\mu')$ for the distance between the origin of μ and the origin of μ'. In rule SR.COM the condition $\mathtt{dis}(\mu,\mu') \leq \mathtt{rad}$ ensures that located processes $\{X(\overline{c})\}_{\mu}$ and $\{Y(\overline{d})\}_{\mu'}$ are close enough to communicate through channel a. The non-stochastic reduction relation of BioScape, $A \rightarrow B$, is defined in Fig. 4. By $\mathtt{translate}(\omega,\mu)$ we denote the function that randomly generates a new map μ', using the movement step ω and the old map μ. The condition $\mu'(\sigma) \subseteq \xi$ of rule NR.MOVE ensures the new located process $\{P[\overline{a}/\overline{x}]\}_{\mu'}$ is within the movement space ξ of X (see previous remark about not checking if the entity moves to an empty space).

For stochastic reductions we compute the duration of the reduction, based on the exponential distribution associated with the propensity of the reduction. Since reductions may have different durations, we introduce *timed configurations*, $\{\!\{A\}\!\}^n$, meaning that, after a time n, this configuration will be A. We extend structural equivalence to timed configurations by adding that $\{\!\{A\}\!\}^0 \equiv A$, and $A \equiv B$ implies $\{\!\{A\}\!\}^n \equiv \{\!\{B\}\!\}^n$. With the metavariables F, and G we denote either spatial configurations or timed configurations (*extended configurations*), i.e.,

$$F,G ::= A \ \mid\ \{\!\{A\}\!\}^n \ \mid\ F \mid G \ \mid\ (\nu a@\mathtt{r},\mathtt{rad}).F \qquad (n \geq 0)$$

We define a reduction strategy that given the whole configuration, first moves all the processes that can be moved, and then executes all the stochastic reductions that can be executed, omitting only reductions which would lead to overlaps, i.e. configurations where some entities occupy the same space. Both non-stochastic and stochastic reductions are applied in parallel. For this purpose, we define multi-hole contexts C by the following grammar:

$$C ::= F \ \mid\ [\,] \ \mid\ C \mid C \ \mid\ (\nu x@r,\mathtt{rad}).C$$

Congruence on multi-hole contexts is naturally induced by the congruence on configuration, associativity and commutativity of the parallel operator, and standard rules for ν restrictions similar to S.NU.COM and S.NU.PAR. Given this congruence any multi-hole context, C, may be written in a *canonical form*. That is, there is C', $C \equiv C'$ such that $C' = \nu_1 \ldots . \nu_n.F_1 \mid \cdots \mid F_m \mid [\,] \mid \cdots \mid [\,]$, where ν_i, $1 \leq i \leq n$, is an abbreviation for $\nu a_i@\mathtt{r}_i,\mathtt{rad}_i$, and for all j, $1 \leq j \leq m$, $F_j = \{\!\{A\}\!\}^n$ for some A, and n, or $F_j = \{P\}_{\mu}$ for some P, and μ. We say that $a_1@\mathtt{r}_1,\mathtt{rad}_1,\ldots,a_n@\mathtt{r}_n,\mathtt{rad}_n$ is $\mathtt{restr}(C)$. In the following we assume that multi-hole contexts are always in canonical form.

As already mentioned, our reduction strategy avoids spatial overlaps. In particular for moving reductions we have to ensure that moves and reshaping are compatible with the available space, that is after moving no entity overlaps with another entity. For stochastic reductions we have to assure that the created entities have their space. To this aim we define the space of a configuration, and a predicate that says whether a configuration does not have any overlapping entities.

Let $\mathtt{space}(F)$ be a function on configuration F that returns the space occupied by its processes located in the global frame defined as follows.

$$\begin{array}{llll} \mathtt{space}(\{P\}_{\mu}) & = \mu(\mathtt{shape}(P)) & \qquad \mathtt{space}(\{\!\{A\}\!\}^n) & = \mathtt{space}(A) \\ \mathtt{space}(F \mid G) & = \mathtt{space}(F) \cup \mathtt{space}(G) & \qquad \mathtt{space}((\nu a@\mathtt{r},\mathtt{rad}).F) & = \mathtt{space}(F) \end{array}$$

We say that a configuration F is *OK* if the various entities in F do not overlap, that is:

$$\textsc{PR.Move}\quad \frac{F_i \rightarrow G_i \quad (1 \leq i \leq p) \qquad C[\underline{G_1}]\cdots[\underline{G_p}]\ \mathrm{OK}_{mv}}{C[F_1]\cdots[F_p] \rightarrowtail C[G_1]\cdots[G_p]}$$

$$\textsc{PR.Timed}\quad \frac{n = min\{n_i \mid 1 \leq i \leq p\} \quad C \text{ is untimed}}{C[\{\!\{A_1\}\!\}^{n_1}]\cdots[\{\!\{A_p\}\!\}^{n_p}] \rightsquigarrow C[\{\!\{A_1\}\!\}^{n_1-n}]\cdots[\{\!\{A_p\}\!\}^{n_p-n}]}$$

$$\textsc{PR.Stoc}\quad \frac{\mathtt{restr}(C) \vdash A_i \xrightarrow{\mathtt{r}_i} B_i \qquad n_i = \tau(\mathtt{r}_i, C_i[A_i]) \quad (1 \leq i \leq p) \qquad C[C_1[\underline{B_1}]]\cdots[C_p[\underline{B_p}]]\ \mathrm{OK}_{st}}{C[C_1[A_1]]\cdots[C_p[A_p]] \dashrightarrow C[C_1[\{\!\{B_1\}\!\}^{n_1}]]\cdots[C_p[\{\!\{B_p\}\!\}^{n_p}]]}$$

$$\textsc{PR.Conf}\quad \frac{F \rightarrowtail F_1 \dashrightarrow F_2 \rightsquigarrow F'}{F \rightarrow F'}$$

Figure 5: Parallel Reduction Relation

$$\{P\}_\mu\ \mathrm{OK} \qquad A\ \mathrm{OK} \Rightarrow \{\!\{A\}\!\}^n\ \mathrm{OK} \qquad F\ \mathrm{OK} \Rightarrow (\nu x@\mathtt{r},\mathtt{rad}).F\ \mathrm{OK}$$
$$F\ \mathrm{OK} \wedge G\ \mathrm{OK} \wedge \mathtt{space}(F) \cap \mathtt{space}(G) = \emptyset \Rightarrow F \mid G\ \mathrm{OK}$$

With the notion of OK configuration we define two notions of *well-formedness of configurations*. The first notion is to be used for parallel move reductions and the second for parallel stochastic reductions. Theses notions are to be used to enforce (i) the fact that only reductions that have enough space for their reduct are allowed, and (ii) that we want *maximal parallelism*, that is any "extra" movement or transformation would produce an overlap. In order to formalise this we first need to single out the sets $\mathfrak{R}_{mv}$ and $\mathfrak{R}_{st}$ of movement and stochastic redexes, i.e. we define:

- $\mathfrak{R}_{mv} = \{\{X(\overline{a})\}_\mu \mid X(\overline{x}) = (\mathtt{mov}.P + M)^{\xi,\omega,\sigma} \in D\}$,
- $\mathfrak{R}_{st} = \{\{X(\overline{a})\}_\mu \mid X(\overline{x}) = (delay@\mathtt{r}.P + M)^{\xi,\omega,\sigma} \in D\} \cup$ $\{\{X(\overline{c})\}_\mu \mid \{Y(\overline{d})\}_{\mu'} \mid X(\overline{x}) = (!a(b).P + M)^{\xi,\omega,\sigma} \in D \;\&\; Y(\overline{y}) = (?a(z).Q + N)^{\xi',\omega',\sigma'} \in D$ $\&\ \mathtt{dis}(\mu,\mu') \leq \mathtt{rad}\}$ where $a@\mathtt{r},\mathtt{rad}$ is the declaration of channel a.

We extend the syntax of configurations by allowing *underlined extended configurations*, defined by: an underlined extended configurations is a configuration in which some spatial sub-configurations may be underlined. Underlined configurations are the tool we use to define maximal parallelism. We can then define:

Definition 1. *(i) An extended configuration F is* OK_{mv} *if F is* OK *and $F \equiv C[A]$ with A not underlined and $A \in \mathfrak{R}_{mv}$ and $A \rightarrow B$ imply $C[B]$ not* OK.

(ii) An extended configuration F is OK_{st} *if F is* OK *and $F \equiv C[A]$ with A not underlined and $A \in \mathfrak{R}_{st}$ and $A \xrightarrow{\tau} B$ imply $C[B]$ not* OK.

As a last notion, we say that a context C is *untimed* if it does not contain timed configurations.

We are now able to explain our parallel reduction strategy, whose rules are given in Fig. 5. The first three rules deal respectively with parallel movements, timed reductions, and stochastic reductions, while the fourth rule maps extended configurations into extended configurations by applying first the parallel movements, then the stochastic interactions, and finally by advancing the time of the minimum required to complete one or more interactions. In this way at the next iteration there would be new entities to be moved and/or stochastically reduced.

The condition of obtaining an OK_{mv} extended configuration in rule PR.Move assures that all possible moves in $C[F_1]\cdots[F_p]$ which do not cause overlaps have been done in the reduction. Similar effect is produced by the conditions that the extended configuration is OK_{st} and that the context is timed in the following two rules, respectively. Rule PR.Stoc prescribes that the time of a stochastic reaction depends (through the function τ) on the rate of the reduction and on the number of available reactants. The outer context C is a multi-hole context, while the context C_i of the redex A_i is a single hole context capturing the surrounding environment that influences the speed of the reduction. We could incorporate a counting function keeping track of the available reactants in the communication range (in a way similar to what is done, e.g., in [2, 6]).

Examples, results of simulations, comparisons with related papers and discussions can be found in the full version of this papers available at `http://www.di.unito.it/~dezani/papers/cdgsst.pdf`.

References

[1] Yifei Bao, Adriana B. Compagnoni, Joseph Glavy & Tommy White (2010): *Computational Modeling for the Activation Cycle of G-proteins by G-protein-coupled Receptors*. In: *MeCBIC'10*, *EPTCS* 40, pp. 39–53, doi:10.4204/EPTCS.40.4.

[2] Livio Bioglio, Mariangiola Dezani-Ciancaglini, Paola Giannini & Angelo Troina (2012): *Typed stochastic semantics for the calculus of looping sequences*. *Theoretical Computer Science* 431, pp. 165–180, doi:10.1016/j.tcs.2011.12.062.

[3] Arne T. Bittig & Adelinde M. Uhrmacher (2010): *Spatial modeling in cell biology at multiple levels*. In: *Winter Simulation Conference*, pp. 608–619.

[4] Luca Cardelli, Emmanuelle Caron, Philippa Gardner, Ozan Kahramanoğulları & Andrew Phillips (2009): *A process model of Rho GTP-binding proteins*. *Theoretical Computer Science* 410(33), pp. 3166–3185, doi:10.1016/j.tcs.2009.04.029.

[5] Adriana Compagnoni, Vishakha Sharma, Yifei Bao, Philippe Bidinger, Livio Bioglio, Eduardo Bonelli, Matthew Libera & Svetlana Sukhishvili (2012): *BioScape: A Modeling and Simulation Language for Bacteria-Materials Interactions*. In: *CS2Bio'12*, ENTCS. To appear.

[6] M. Coppo, F. Damiani, M. Drocco, E. Grassi, E. Sciacca, S. Spinella & A. Troina (2012): *Simulation Techniques for the Calculus of Wrapped Compartments*. *Theoretical Computer Science* 431, pp. 75–95, doi:10.1016/j.tcs.2011.12.063.

[7] Daniel T. Gillespie (1977): *Exact stochastic simulation of coupled chemical reactions*. *The Journal of Physical Chemistry* 81(25), pp. 2340–2361, doi:10.1021/j100540a008.

[8] Mathias John, Roland Ewald & Adelinde M. Uhrmacher (2008): *A Spatial Extension to the π Calculus*. *ENTCS* 194(3), pp. 133–148, doi:10.1016/j.entcs.2007.12.010.

[9] CM Macal & MJ North (2010): *Tutorial on agent-based modelling and simulation*. *Journal of Simulation* 4(3), pp. 151–162, doi:10.1057/jos.2010.3.

[10] Nicolas Le Novère & Thomas Simon Shimuzu (2001): *StochSim: modelling of stochastic biomolecular processes*. *BioInformatics* 17(6), pp. 575–576, doi:10.1093/bioinformatics/17.6.575.

[11] Francesco Pappalardo, Pierluigi Lollini, Filippo Castiglione & Santo Motta (2005): *Modeling and simulation of cancer immunoprevention vaccine*. *Bioinformatics* 21(12), pp. 2891–2897, doi:10.1093/bioinformatics/bti426.

[12] Andrew Phillips & Luca Cardelli (2007): *Efficient, Correct Simulation of Biological Processes in the Stochastic Pi-calculus*. In: *CMSB'07*, *LNCS* 4695, Springer, pp. 184–199, doi:10.1007/978-3-540-75140-3-13.

[13] Nicholas F. Polys, Doug A. Bowman, Chris North, Reinhard Laubenbacher & Karen Duca (2004): *PathSim visualizer: an Information-Rich Virtual Environment framework for systems biology*. In: *Web3D'04*, ACM Press, pp. 7–14, doi:10.1145/985040.985042.

[14] Corrado Priami, Aviv Regev, Ehud Y. Shapiro & William Silverman (2001): *Application of A Stochastic Name-passing Calculus to Representation and Simulation of Molecular Processes*. *Information Processing Letters* 80(1), pp. 25–31, doi:10.1016/S0020-0190(01)00214-9.

[15] Nicolas Rapin, Ole Lund & Filippo Castiglione (2011): *Immune System Simulation Online*. *Bioinformatics*, doi:10.1093/bioinformatics/btr335.

[16] A. Stefanek, M. Vigliotti & J. T. Bradley (2009): *Spatial extension of stochastic pi calculus*. In: *8th Workshop on Process Algebra and Stochastically Timed Activities*, pp. 109–117.

[17] David S. Wishart, Robert Yang, David Arndt, Peter Tang & Joseph Cruz (2005): *Dynamic cellular automata: an alternative approach to cellular simulation*. *In Silico Biology* 5(2), pp. 139–161.